HD28
.M414
no. 3481-
92
1993

DEWEY

THE HYPERCUBE OF INNOVATION

Allan N. Afuah
Massachusetts Institute of Technology

Nik Bahram
Digital Equipment Corporation

November 1992
Revised July 1993

WP #3481-92 BPS

Forthcoming *Research Policy*

THE HYPERCUBE OF INNOVATION

Allan N. Afuah
Massachusetts Institute of Technology

Nik Bahram
Digital Equipment Corporation

November 1992
Revised July 1993

WP #3481-92 BPS

Forthcoming *Research Policy*

© Massachusetts Institute of Technology

Massachusetts Institute of Technology
Sloan School of Management
50 Memorial Dr.
Cambridge, Massachusetts 02139

M.I.T. LIBRARIES
SEP 9 1993
RECEIVED

The Hypercube of Innovation.

Abstract

Innovation has frequently been categorized as either radical, incremental, architectural, modular or niche, based on the effects which it has on the competence, other products, and investment decisions of the innovating entity. Often, however, an innovation which is, say, architectural at the innovator/manufacturer level, may turn out to be radical to customers, incremental to suppliers of components and equipment, and something else to suppliers of critical complementary innovations. These various faces of one innovation at different stages of the innovation value-adding chain are what we call the hypercube of innovation. For many high-technology products, a technology strategy that neglects these various faces of an innovation and dwells only on the effects of the innovation at the innovator/manufacturer level, can have disastrous effects. This is especially so for innovations whose success depends on complementary innovations, whose use involves learning and where positive network externalities exist at the customer level.

We describe the hypercube of innovation model and use it to examine RISC (Reduced Instruction Set Computers) and CISC (Complex Instruction Set Computers) semiconductor chips, and supercomputers, and suggest how firms can better manage the relationships along the innovation value-adding chain using the model. The model forces innovation managers to think in terms of their customers, suppliers and complementary innovators.

Acknowledgments

The authors wish to thank Rebecca Henderson, Lotte Bailyn, Eric Rebentish, and two anonymous referees for their valuable suggestions. Any errors or omissions, however, remain the responsibility of the authors.

1. Introduction

Ever since Schumpeter, scholars of innovation, in an effort to better understand how to manage the process of innovation, have tried to categorize innovations as a function of what the innovations do to the skills, knowledge, investment strategies, and existing products of the innovating entity. But these categorizations of innovations have had one main drawback: By choosing to concentrate on the effects of the innovation on the competence of the innovating entity, they have neglected the effects of the innovations on the competence and assets of suppliers of key components and equipment, customers, and suppliers of complementary innovations. But the fact is that, more often than not, an innovation that is, say, incremental at the innovator/manufacturer level, may turn out to be radical to customers and something else to suppliers of critical complementary innovations; all of which have implications for the success of the innovation. These various faces of one innovation at different stages of the innovation value-adding chain are what we call the hypercube of innovation.

Schumpeter himself described innovation as "a historic and irreversible change in the way of doing things" and "creative destruction" [Schumpeter, 1947]. Abernathy and Utterback (1978) found that as a technology evolves, product innovation gives way to process innovation making it difficult for the innovating entity to revert to new product innovations; that is, the competence of the innovating entity is effectively destroyed. Using the automobile industry, Abernathy and Clark(1986) grouped innovations into four categories depending on the impact of the innovation on the innovating firm's capabilities and knowledge of its technology or market. They didn't address the impact of each of the innovations on the capabilities and assets of their suppliers of components, customers, and suppliers of complementary products. Using extensive data from the photolithography industry, Henderson and Clark (1990) classified innovations according to whether the innovation overturned the existing knowledge of core concepts and components, and the

linkages between them. They classified an innovation as radical if the core concepts of the innovation as well as the linkages between them overturned existing ones; architectural if the core concepts were being reinforced while the linkages between these core concepts and components of the product were changed; incremental if the core concepts were reinforced while the linkages between them were unchanged; modular if the core concepts were overturned while the linkages between the concepts were unchanged; radical if the core concepts and linkages between them are overturned. As was the case with the Abernathy and Clark analysis of the automobile industry, the impact of the innovations on the capabilities and assets of suppliers, customers and suppliers of complementary products was not considered. In their study of the U.S. cement, and minicomputer industries, Tushman and Anderson (1986), and Anderson and Tushman (1990) classified innovations as "competence destroying" or "competence enhancing" depending on what the innovation did to the knowledge base of the innovating entity.

Roberts and Berry (1985), prescribed how an innovating entity can enter a new business depending on it's familiarity with the technology and market, and the newness of the market and technology to the innovating entity. These business entry options range from acquiring other firms with the technology and market competence or performing the R&D internally for the familiar, to venture capital vestments for the unfamiliar and new. The familiarity of the technology to members of the innovation value-added chain was not investigated.

Given the nature of some of the industries studied by these authors, the conclusions they arrived at vis-a-vis what the innovating entity should do, shouldn't differ much from the conclusions that would be arrived at from an analysis that uses the hypercube model. However, for industries where at least one of the following is true: complementary innovations are critical to the diffusion and success of products; where learning by customers is critical, expensive and often results in lock-in; where positive network externalities at customers are common and where equipment and critical components (that

go into the innovation) from suppliers can be innovations in their own right; an examination of the effects of an innovation cannot be limited to the impact on the capabilities, competence and assets of the innovating entity. An analysis must also look at the impact of the innovation on the capabilities of suppliers of components, customers, and complementary innovators.

Stated differently, studies that have categorized innovation, have the innovating entity asking the question: "What is the impact of this innovation on my organizational capabilities, competence, existing products, knowledge of components, key concepts and linkages between them". In the hypercube of innovation approach we are suggesting that in addition to probing what the innovation will do to its competence and assets, the innovating entity must also ask the question: "What will my innovation do to the competence and products of my suppliers, OEM customers, end-user customers, and suppliers (some of which are competitors) of key complementary innovations—that is, what is the impact of the innovation at the various stages of its value-added chain?"

The hypercube model forces innovation managers to think in terms of what the impact of their innovation is going to be on customers, suppliers of critical components, equipment, and complementary innovations. Since customers and users can also be future innovators (von Hippel, 1988), the hypercube may also help the innovating entity track potential competitors and complementary innovators.

In this paper we describe and illustrate the innovation hypercube model using anecdotal examples from different industries and then use it to examine the RISC (Reduced Instruction Set Computers), CISC (Complex Instruction Set Computers) semiconductor microchips, and supercomputers industries. From the examination, we suggest some measures that the innovating firm could take to avoid getting lost in the cube. RISC, CISC and supercomputers are particularly interesting examples for various reasons: They depend on complementary innovations for market success, exhibit positive network

externalities at customers, require complex equipment and components from suppliers, and their use often involves a lot of learning.

2.0 The Hypercube Model.

For this model, we will focus on product innovation as the unit of analysis. The product—the final output of the innovating entity—needs critical components or high tech equipment as inputs and can be sold directly to an end-user or sold to an OEM (original equipment manufacturer) who adds value to it and then resells it to end-users. The product also possess some subset of the following: 1) it requires some considerable skill or knowledge to use or maintain which can be obtained by learning. 2) The value of the product to the owner increases as more people own the product, i.e., it possesses positive network externalities (David, 1985; Kartz and Shapiro, 1985). 3) Complementary innovations are critical to diffusion and use of the innovation.

As we stated earlier, an innovation that is architectural to the innovating entity may be radical to customers and suppliers, and incremental to complementary innovators. The hypercube of innovation model examines these different faces which an innovation assumes at the different stages of the innovation value-added chain—the innovating entity, suppliers, customers and complementary innovators—and suggests how the innovating entity can best deal with them. It looks at the impact that an innovation has, not only at the innovating entity, but also at suppliers of components, OEM customers, end-users, and complementary innovators. It depicts relationships that are multidimensional in nature. In particular, the hypercube is a 4-dimensional cube with each of the stages of the innovation value-added chain representing a dimension, and the location of any innovation in this 4-dimensional space being determined by the "intensity" of the innovation along each of these dimensions; where intensity is a measure of how radical the innovation is, using an ordinal scale, say, of incremental = 1, modular = 2, architectural = 3, and radical = 4, with

intensity increasing from incremental to radical. Because of the difficulties in visualizing things in four dimensions, however, we have transformed the 4-dimensional hypercube to the 3-dimensional cube of Figure 1 (and later, to the two dimensional GREEN-RED zone map). In Figure 1, the transformed hypercube (shown as a parallelepiped) has a cross-section (X and Y axes) that categorizes innovations according to their impact on the capabilities, competence, assets, and products of the actor in question, and a length on which are located the different actors in the innovation value-adding chain, viz.: suppliers of critical components that go into the innovation, customers [OEM (original equipment manufacturer), and end-user], and suppliers of complementary innovations. Figure 2 further explodes these stages of the chain for better visualization. We emphasize the fact that the innovating entity can use any criteria for categorizing innovations in the X and Y axis.

Tables 2.1, 2.2, 2.3 and 2.4 below list the range of possible impact of an innovation at the innovating entity, suppliers, customers, and complementary innovators, and we briefly describe what is possible at each stage.

2.1 The Innovating Entity

The focus of most innovation literature has been on the impact of an innovation on the capabilities and assets of its innovator. For the innovator, the primary concern has been the impact of the innovation on its organizational competence—whether it enhances or destroys it [Abernathy and Utterback, 1978; Tushman and Anderson, 1986]—on the core concepts and linkages between those core concepts of the product [Henderson and Clark, 1990], on existing innovations, and on the willingness of management to invest in the innovation [Henderson, 1993; Reinganum, 1983, 1984]. For the hypercube model, any categorization framework can be used. For example, we could use the Henderson and Clark [1990] model and classify an innovation as radical, if the core concepts of the innovation as well as the linkages between them have overturned existing ones; architectural, if the core concepts are being reinforced while the linkages between these core concepts of

the product are changed; incremental, if the core concepts are reinforced while the linkages between them are unchanged; modular, if the core concepts are overturned while the linkages between the concepts are unchanged. However, for simplicity, we will use the dichotomy of incremental and radical innovations.

As detailed in Table 2.1 below, the innovating entity has to recognize and take the necessary corrective action depending on whether the innovation makes obsolete or enhances previous designs, destroys or enhances knowledge gained in previous designs, cannibalizes older products, can be used with previous complementary products.

THE INNOVATING ENTITY	
Asset or Activity	***Possible Impact***
Core Concepts	Enhances or makes obsolete core concepts from previous innovations
Linkages between core concepts and components	Enhances or destroys previous knowledge of linkages
Product Components	Remain the same or change
Competence	Enhances or destroys other competencies (skills & knowledge from previous innovation)
Existing products	Enhances use of previous products or cannibalizes them
Complementary innovations from previous products	Can use or not use complementary innovations
Institutional support	Can receive or not receive any government or other institutional research subsidies

Table 2.1: The effects of the Innovation on the Innovating Entity

2.2 Customers

The impact of an innovation on the capabilities and assets of the innovator's customers has very important implications for the market success of the innovator. Unfortunately, most innovation studies have focused on the impact of the technology on the innovator's knowledge of technology and market, while ignoring the impact on customers' capabilities and assets. There are at least four areas where the impact of an innovation on a customer can have serious effects: Learning, positive network externalities, compatibility with complementary or old products and continued use of old products.

2.2.1 Learning: Many complex high-technology products require that users invest time and money in learning how to operate and maintain the products. An innovation that destroys

the knowledge that the customer has acquired, has a smaller chance of being adapted than one that enhances this knowledge and skills. Thus we expect a person who buys a computer and learns the computer's operating system, to be less willing to buy another computer with a different operating system than one with the same operating system; unless there is another program that can make the new operating system transparent to the old user.

2.2.2 Positive network externalities: A product or skill is said to possess positive network externalities if the value of the product to an owner increases as more people own it. Positive network externality has its origins from the telephone network where one's telephone is more valuable the more people are connected to one's network. The more friends you have that own a computer that is compatible with yours, the more valuable your computer is to you because you can share software and innovative ways of using the computer. An innovation that destroys this positive network externality does not stand a good chance of being adapted by customers.

2.2.3 Compatibility with complementary products: Using the computer example again, a personal computer user who invested in a Lotus 123 spreadsheet would prefer not to switch to a new computer that requires him to buy a new spreadsheet.

2.2.4 Built-up assets: An airline that has built maintenance facilities for Boeing 737s but has to change to a fleet of Airbus A320s, will have problems with the new parts inventory and maintenance procedures that must now replace the old one.

A user who has written her own applications programs to run a Macintosh will not be easily convinced to switch to an IBM personal computer, if his Macintosh programs cannot run on the new machine.

From all these, it is evident that the innovating entity must make sure that 1) the innovation will not destroy the skills and knowledge that its customers learned with previous innovations, 2) it will not destroy any positive network externalities that previous innovations may have created for customers, 3) customer's complementary products can still be used with the new innovations, 4) built-up assets will not have to be destroyed.

CUSTOMER	
Asset or Activity	*Possible Impact*
Learning	Enhances or destroys skills & knowledge acquired from previous product.
Built-up assets	Enhances or destroys use of assets built around previous innovations
Network externalities	Enhances or destroys positive network externalities
Complementary innovations from previous products	Can use or not use complementary innovations from older products
Product Design	Enhances or makes obsolete previous design.
Design knowledge	Enhances or destroys previous design knowledge
Product Components	Remain the same or change

Table 2.2: The effects on Customers

2.3 Complementary Innovators

The huge success of personal computers since their introduction in the late seventies would not be as phenomenal were it not for complementary innovations like spreadsheet and word-processing software. Innovators not only have to watch out for the inertia of older complementary innovations and the momentum of newer ones, but may also have to cooperate (via, e.g., strategic alliances) with the complementary innovators to produce complementary innovations (see the case of IBM and Intel's microprocessor later).

COMPLEMENTARY INNOVATORS	
Asset or Activity	*Possible Impact (Range—Best to worst)*
Inertia of old complementary products	Keeps up with inertia of old complementary innovations
Momentum of new products	Keeps up with the momentum of new complementary innovations
Product Design	Enhances or makes obsolete previous design (of complementary product).
Design knowledge	Enhances or destroys previous design & manufacturing knowledge of complementary product
Product Components	Remain the same or change
Competence	Enhances or destroys skills & Knowledge from previous products
Existing products	Enhances use of previous products
Positive Network externalities	Enhances or destroys positive network externalities for complementary products
Complementary innovations from previous products	Can use or not use complementary innovations from older products

Table 2.3: The effects of the Innovation on Complementary Innovators

2.4 Suppliers of components and equipment

Some high technology product innovations (e.g., aircraft and supercomputers) depend heavily on component and equipment innovations from their suppliers. The aircraft cannot move into supersonic flight without the right innovations in engine technology. In supercomputers, most of the gains that we have seen in computer performance have come from innovations in the semiconductor chips that go into them.

SUPPLIERS of COMPONENTS and EQUIPMENT	
Asset or Activity	*Possible Impact (Range—Best to worst)*
Component and/or equipment Design	Enhances or makes obsolete previous design of component or equipment supplied for previous innovation
Design Knowledge of components and/or equipment	Enhances or destroys previous design and manufacturing knowledge of components or equipment supplier for previous innovation
Competence	Enhances or destroys skills & knowledge used to supply components or equipment for pervious innovation.
Old products	Enhances or destroys use of previous components or equipment.

Table 2.4: The effects of the Innovation on Suppliers of Key components or equipment.

2.5 The GREEN-RED zone map

The *Green -Red Zone* map of Figure 3 is a simplified two-dimensional version of the hypercube. It is a map of the different faces that an innovation assumes at the different stages of the innovation value-added chain. In the figure, the effect of Innovation A is incremental on suppliers, radical on the innovator, modular on customers, and incremental on complementary innovators. The green zone is where innovations reinforce core concepts, skills and knowledge, and an innovation that falls in this zone for the innovator, supplier, customer and complementary innovators, can be very attractive to the innovating entity.

The red zone covers the area where previous core concepts are overturned, and competence destroyed at the various stages of the chain. This is the zone for radical innovation. Any innovation whose map passes through this zone, especially at the customer level, should not be pursued unless a subset of the following is true: 1) The price/performance ratio of the innovation, as viewed by the customer, outweighs any losses incurred as a result of competence or positive network externality destruction. This

happens, for example, when the physical limit of an older technological trajectory has been reached and the only way to overcome this physical limitation is to move to a new technological trajectory—a move that often means destruction of competence acquired during the evolution along the older trajectory but substantial improvement in some key parameter For example, parallel computers as a result of the physical limits reached by microprocessor technology. 2) New markets where customers have not yet had time to build any innovation-specific skills and knowledge, and competence destruction is not an issue. For example, the adoption of DOS in the PC market. 3) Complementary innovations, that allow customers to keep their competence and positive network externalities, exist. An example of such a complementary innovation would be a software package designed to allow PC users who are only familiar with DOS to be able to sit at a Macintosh and use DOS as they would on a PC, making the Macintosh operating system transparent to the user so that customers' competence is not destroyed when a customer moves from one machine the other. 4) When institutional requirements mandate the innovation. Electric cars for LA are an example.

Referring again to Figure 3, Innovation A may present the innovator with more problems than Innovation B since A's map along the innovation value-added chain passes through the RED zone while B's doesn't.

2.5.1 Cost-benefit analysis

Assessing the cost and returns on investments (ROI) should not be limited to the innovating entity. Innovation managers should explicitly quantify the costs and investment requirements, carefully weighing them against the benefits at all levels of the chain. At the customer level, the price/performance benefits must be weighed against the loss in network externality, additional investment in learning and idiosyncratic complementary assets. At the supplier level, costs include new development and production processes, retooling,

learning, and obsolescence of existing production capacity. The innovating entity should make similar cost-benefit analysis for the complementary innovator level.

The hypercube of innovation concept is best illustrated with examples such as the DSK (Dvorak Simplified Keyboard) keyboard, the electric car, IBM's OS-2 operating system and Microsoft's Windows.

2.6.1 The DSK Keyboard

The DSK (Dvorak Simplified Keyboard) is an example of an innovation that was architectural to the innovating entity, incremental to suppliers of components and complementary products but radical to customers. Figure 4a shows the GREEN-RED zone map of DSK.

DSK is a keyboard arrangement that by many estimates allowed people to type 20-40% faster than with the QWERTY arrangement that most of today's keyboards have. But by the time the DSK innovation was being marketed, the QWERTY keyboard had been adapted by many customers who learned how to type with it [David 1985]. Switching from QWERTY to DSK meant two things to the customer who had already learnt to type with the former: 1) He/she would have to learn how to type again effectively abandoning the old skills and knowledge of QWERTY. 2) He or she would have a smaller market for his/her skills since more potential employers needed people with QWERTY typing skills. Customers who didn't know how to type at all realized that the QWERTY skills would be more valuable to them since more people and more places of employment use the QWERTY keyboard arrangement. This phenomenon where a product or skill is more valuable the more people that have it, is called positive network externality. So, to potential customers, adoption of the DSK would destroy their competence and/or positive network externalities, and therefore constitutes a radical innovation.

To the innovators of DSK this was an architectural innovation since the core concepts and components for the keyboard had not changed but the linkages between them had changed since the keys were arranged different.

To suppliers of components or complementary products, DSK had no impact on their skills, and products.

There may be other reasons why the DSK keyboard failed to displace QWERTY despite the former's superior performance, but the fact that this innovation was radical to customers has to be a key one.

Figure 4a shows the map through the value-added chain for the DSK keyboard.

2.6.2 The Electric Car

The electric car is still under development. But we can speculate, for illustrative purposes, on what the innovation's impact is on the innovation value-added chain. This is a radical innovation to the car companies, to suppliers of key components like the power train, and to suppliers of the key complementary innovation--gasoline. But to customers, it will be an incremental innovation. The GREEN-ZONE map is shown in Figure 4b. What we know as the power train—engine, transmission, fuel injection, and exhaust system—of the gasoline-powered automobile is being replaced by the an electric motor, battery and electric motor controller in the electric car [see for example, Pratt, 1992].

Thus, not only are the key components and design concepts for the electric car different from those of the gasoline-powered car, the linkages between them are also different. For gasoline-powered car manufacturers, development of the electric car is a radical innovation. To suppliers of the power train components for gasoline-powered cars, the electric car destroys a lot of their competence, and is also a radical innovation to them. The electric car also runs on electricity, not gasoline, and so to gasoline companies, the electric car is also a radical innovation. To customers, however, it is an incremental innovation, since drivers of gasoline-powered cars can keep their driving skills, and other knowledge of operating cars, but get a car that emits less pollution. They may have to throw away that old container for gasoline.

2.6.3 OS-2 and Windows

When IBM and Microsoft, two of the largest beneficiaries of the PC and PC compatible market, found out how popular the "look and feel" of the Apple Macintosh personal computer was becoming, they decided to develop an operating system with a similar "look and feel" called OS-2 for IBM PCs and PC-compatibles. OS-2 would also offer many advantages over DOS including multitasking (have the computer run more than one applications program at any one time). To both firms this was a radical innovation as core concepts would have to be changed to support multitasking and other key factors. To most customers of the IBM PC and PC compatibles who had already learned to use the DOS operating system and had seen the advantages of Icon- and windows-based user interface of the Apple Macintosh, OS-2 would be an incremental innovation. This was particularly true since all DOS applications would run under OS-2.

Faced with the daunting challenge of a radical innovation Microsoft and IBM parted ways with IBM advocating the investment in making the OS-2 radical innovation and Microsoft favoring a more incremental innovation. Microsoft's strategy led to the creation of Microsoft Windows several years in advance of IBM's introduction of OS-2. Although OS-2 is a technically better product, the revenues generated by Microsoft Windows and the ensuing increase in shareholder value suggests that Microsoft's approach may have been the more successful of the two. Microsoft has since put plans in place to enhance Microsoft Windows to Microsoft Windows NT which will have similar functionality as IBM's OS-2. Meanwhile, Microsoft Windows has a 10 to 1 advantage over OS-2 in installed base. The map of both innovations through the innovation value-added chain is shown in Figure 5.

Mapping Innovations into the Hypercube

Figure 6 explodes the hypercube to show cross-sectional slices of the cube at each stage of the innovation value-added chain. It shows where the innovation of the DSK

keyboard by Dvorak, a keyboard designer/manufacturer, and of the electric car by gasoline-powered car designer/manufacturer, fit on the innovation hypercube.
The DSK innovation is architectural to its innovator, incremental to suppliers of components and complementary innovations but radical to customers. Each face of the hypercube or stage of the value added chain is shown in figure 6 with the kind of innovation as perceived at that stage. The electric car innovation is radical to the gasoline-powered car manufacturers, suppliers, and complementary innovation suppliers, but incremental to customers of the cars. OS-2 was radical to IBM, Microsoft Windows was incremental to Microsoft and both were architectural innovations to complementary innovators like Lotus and most importantly, incremental to customers.

2.7 Summary of the model

An innovation that is incremental to the innovating entity may be radical to customers, and something else to complementary innovators and suppliers of critical components for the innovation, and a technology strategy that dwells only on the impact of an innovation on the innovating entity, may be in for disastrous consequences. The hypercube model forces managers of the innovating entity to examine the impact of their innovation at all the stages of the innovation value-added chain. The model suggests that innovations that reinforce core concepts and enhance competencies along the innovation value-added chain should be pursued. Those that destroy competence, positive network externalities, and assets at any stage of the chain, especially at customers, may provide the unwary innovator problems. The map of such an innovation passes through the red zone (see figure 3). Such innovations should be avoided except where there are obvious price/performance advantages for the customer; the innovator is entering new markets where customers have not yet had time to build any innovation-specific skills and knowledge, and competence destruction is not an issue; complementary innovations, that allow customers to keep their competence and positive network externalities, exist; and

when institutional requirements mandate the innovation. Somewhere between these two extremes is the yellow zone. Any innovation whose map passes through this zone should be pursued with a lot of precaution. The innovator should monitor the inertia of older complementary innovations and the momentum of newer ones, to take advantage of them.

Finally, the innovating entity should perform the relevant cost-benefit analysis for each level of the innovation value-added chain, taking into consideration not only the cost of learning, network externalities, additional capital investments, and cannibalization of old products, but also of such additional expenses as marketing and advertising for the new technology.

We are now ready to apply the model to RISC, CISC and supercomputers.

The Hypercube: The cases of RISC, CISC, and Supercomputers.

3.0 RISC & CISC Chips

CISC (Complex Instruction Set Computer) chips:

In 1970 Intel Corporation invented the first microprocessor, a microchip implementation of the Central Processing Unit (CPU) of a computer. Subsequently, Intel and its competitors like Motorola introduced successive generations of 8-bit, 16-bit and 32-bit microprocessors that over time, got faster, consumed less power, and delivered higher functionality. The instruction sets for these processors--the commands which programmers use to tell the processors what to do--grew to be very large, with each instruction taking too long to execute, and earning these processors the name Complex Instruction Set Computers (CISC).

These relatively complicated instructions required ever increasing chip sizes, cost more, and placed a limit on how fast a given operation could be executed, limiting the microprocessor's speed.

Microprocessors differ from other chips in that they have instruction sets, and development systems/software have to be written to support the instruction sets. They also require a host of compatible complementary chips to allow them interface with devices like printers, modems or keyboard, and chips that control disk drives. These complementary chips and development systems can also be supplied by complementary innovators.

When Intel designed its third generatior microprocessor (16-bit), the 8086, it discovered that a lot of the less expensive complementary chips in existence then were for the previous generation of microprocessors, i.e. 8-bit microprocessors. Intel had two choices: wait until the complementary chips catch up with it's third generation microprocessor or innovate again. Intel chose the latter and designed the 8088 with the 16-bit internal architecture of the 8086, and the external architecture of 8-bit processors so that the 8088 could use the readily available and inexpensive 8-bit complementary chips. This allowed Intel's customers, i.e. system builders like IBM, to take advantage of the advanced third generation features that the internal architecture provides while also using the inexpensive, more readily available second generation complementary chips.

When IBM decided to enter the Personal Computer (PC) market, and had to choose a microprocessor for its PC, it chose the Intel 8088 although this processor may not have been superior to other microprocessors, specifically the Motorola 68000. The 8088's 8-bit interface that allowed PC manufacturers to use readily available and inexpensive complementary chips may have tilted the balance in Intel's favor.

RISC (Reduced Instruction Set Computers) Chips:

By the mid 1970s industry researchers and academics had begun to question the efficiency of the CISC approach. Alternative approaches to remove the performance and cost disadvantages inherent in the CISC approach, were examined. In 1975, IBM's Thomas J. Watson Research Center's researchers began the development of the IBM 801

computer. Although not a microprocessor, this computer laid out the foundations of the Reduced Instruction Set Computer (RISC) approach to microprocessor design. These IBM researchers moved away from a large number of complicated instructions to a small number of very simple instructions. This change significantly enhanced the speed of the processor without severely impacting the ease of use and flexibility that was advertised as a key advantage of CISC processors.

The simpler and fewer instructions meant smaller chip sizes and faster RISC processors. Successive generations of RISC processors could be designed faster than same generation CISC processors providing RISC with a Time-To-Market (TTM) advantage. These factors would lead to large RISC price/performance ratio advantages over same generation CISC processors.

By 1982, Patterson et al at Berkeley, and Hennessey et al at Stanford had implemented the RISC concepts in single VLSI (Very Large Scale Integrated) circuits. The RISC1 processor at Berkeley, and the MIPS processor at Stanford laid the foundation of the two most successful RISC architectures in the industry today: SUN Microsystem's SPARC RISC processor and MIPS Corporation's family of Rxxxx (R2000, R3000, R4000) processors.

Despite pursuing the RISC principles first, IBM has only recently developed a relatively successful VLSI RISC processor of its own, the RS6000, and plans on co-developing the Power series of RISC processors with Motorola and Apple Computer.

Digital Equipment Corporation, at first used RISC chips from the MIPS family of processors to power its DECStation workstation products, but has recently introduced its 64-bit Alpha microprocessor which will be used for Digital's products ranging from Desktop to Data Center minis and mainframes.

Hewlett Packard has also moved into adopting RISC for its hardware platforms as well. The PA-RISC architecture is now available across the range of HP's hardware platforms.

In the next two sections the hypercube of innovation is used to examine innovations in RISC and CISC microporcessors. Figure 7 provides the context for RISC and CISC use.

The Hypercube model and CISC.

In Figure 8 we have classified some CISC processor innovations according to whether they were incremental or radical innovations, and below, we look at the effects of some of these innovations at the various stages of the innovation value-added chain.

Integrating the CPU of a computer on a single chip, the microprocessor, in 1970 was a radical innovation to Intel. The design methodologies were not only different, semiconductor fabrication capability had to be pushed to its limits to achieve the desired integration levels. The demands put on suppliers of critical equipment like CAD tools, were radical. To OEM customers, this was also a radical innovation since they had to learn the instruction sets of these processors, etc. For complementary innovators, this was a radical innovation since they had to develop new compatible complementary chips for the microprocessors while also learning the instruction sets of the processors. They had to develop systems/software to support the microprocessor's instruction set.

Design and implementation of subsequent generations in the same microprocessor family, for example Intel's 80xxx family of CISC processors, are for the most part incremental innovations to the various members of the innovation value-added chain, partly as a result of the fact that microprocessors were designed to be upward compatible with previous generations, allowing programs that were written for old processors to work on new systems. Thus programs written for Intel's 16-bit microprocessors could run on 32-bit processors. Complementary innovators like Microsoft who wrote the DOS operating system used on 16-bit machines didn't have to worry about writing a new operating system because DOS could be used for the 32-bit machines.

The Hypercube Model and RISC

Figure 8 illustrates the hypercube applied to various RISC processors from key innovators.

RISC processors utilize the same building blocks as CISC processors, but the way these blocks are put together is different for each processor type. As such, RISC compared to CISC is an architectural innovation (Henderson and Clark, 1990) for the innovating entity.

For many complementary innovators, the first generation of complementary innovations in support of RISC were for the most part radical since in many cases, many changes were required in products used with CISC microprocessors. Compilers used by RISC offers a good example. Fundamental to RISC hardware simplicity is more sophisticated compiler technology that must translate "friendly" programming languages such as FORTRAN or COBOL into a reduced and simplified set of microprocessor instructions that the RISC processor can understand. The availability of such optimized compilers is often cited as one of the reasons why the RISC dream was realized. To providers of compilers, the first generation of RISC chips was a radical innovation. To developers of Development Systems, RISC was also a radical innovation since they now had to learn the new instruction sets of RISC, and also had to find ways to accommodate RISC's higher speeds and inherent parallelism. Complementary ICs were hard pressed to keep up with the RISC microprocessor speed. Chips that decouple other slower parts of the system from the faster microprocessors were developed. Most of these innovations rely on rearranging the linkages between existing building blocks, and as such, most of the innovations to date have been architectural in nature.

Once again, most of these innovations were adopted by CISC processors and have become standard design practice for microprocessor based system design.

For the most part, the silicon fabrication technology and CAD tools that suppliers provided CISC makers can be used by RISC users too. This does not mean that there have been no radical or architectural changes in these fields. There have been many. The point is

that these innovations have been independent of RISC and CISC architectures. The impact of these architectures has been to create incremental innovation in the supplier base.

For RISC end-users, the picture has been very different and varied. They have in general adopted the UNIX operating system, making RISC a radical innovation for those who had used other operating systems like DOS. But the advantages of a non-proprietary, very low cost, high performance and portable operating system coupled with the superior price/performance of RISC microprocessors, has for most applications, made up for the loss of competence in switching operating systems. To the end-user, porting applications software from non-UNIX operating systems to UNIX is also costly.

For relatively new markets like workstations or embedded control, where there were no entrenched operating systems and where price/performance is critical, RISC has been accepted relatively fast.

For markets like personal computers where end-users have not only invested heavily in learning, and writing their own applications software in DOS-based systems, but also built-up positive network externalities, RISC is a lot more of a radical innovation than for the workstation or embedded control market. Thus RISC has had great difficulties dislodging CISC. The weak appropriability (Teece, 1986) of RISC concepts has helped CISC designs to close the gap in price/performance.

Later generations of RISC, like MIPS Corporation's R3000, and R4000 or later versions of IBM's RS6000, have so far been mostly incremental innovations for all members of the innovation value-added chain.

4.0 Supercomputers:

Supercomputers are generally described as the most powerful computational systems available at any given time. This would mean that the first supercomputer dates back to Charles Babbage's mid 1800s "analytical engine". Most of today's installed base of supercomputers, however, can be attributed to Seymour Cray who, in

1975, left Control Data Corporation (CDC) where he had designed supercomputers, to start his own supercomputer company, Cray Research Inc.. At CDC, Cray had designed the CDC 7600 supercomputer, a so-called scalar supercomputer because it had a scalar processor (the "engine" or brain of the computer). Scalar processors have to issue an instruction for every single operation (e.g. addition of two numbers) so that even vector data would have to be broken down and an instruction issued for operation on each element of the vector. The 7600 was also of the traditional Von Neumann architecture[1]. In 1976, Cray Research shipped its first supercomputer, the Cray-1, the first commercially available vector supercomputer. Vector computers, for the most part, need only one instruction to execute each operation on vectors, and this greatly improves processing time (for applications that lend themselves to lists) compared to scalar processors. Vector processing was a key innovation in supercomputers especially since a lot of data on which supercomputers operate are either vector-like or could be vectorized. The first vector supercomputer was actually the CDC Star-100 but was not commercially available until after the Cray-1 when it was released as the Cyber 205.

One thing which the Cray-1, CDC7600, Cyber 205, and previous supercomputers (vector or scalar) had in common was that they each had only one processor that could be put on any one processing job at any one time. Cray Research changed all that in 1982 when it introduced its multiprocessor Cray X-MP, the first commercially successful supercomputer to apply more than one processor to the same problem at any one time (The ILLIAC IV, developed at the University of Illinois, was the first parallel supercomputer). In the years that followed, Cray Research introduced many other multiprocessor supecomputers with the Cray Y-MP C90 its latest with 16

[11]The architecture used in most of today's computers is often attributed to John Von Neumann's mid-1940s architecture. In that architecture, the Central Processing Unit (CPU) of the computer fetches an instruction (data) from a central store (main memory), operates on it (for example, add or subtract), and returns the results into the main memory. Only one CPU is used, and that one CPU can do only one thing at a time.

processors in 1991 that delivers 16 GFLOPS (gigaFLOPS = billion floating point operations per second) compared to the Cray-1's 100 MFLOPS (million floating point operations per second). In 1992, NEC introduced its 4-processor SX3 that gives 25 GFLOPS. Table 10 lists some of the key supercomputers that have been introduced over the years.

Most of the gains in supercomputer performance have come as a result of innovations in semiconductor technology, from the transistor to very large scale integrated (VLSI) circuits. NEC's four-processor supercomputer, for example, was able to deliver the 25 GFLOP primarily because of its advanced ECL (emitter-coupled Logic) semiconductor technology and premier packaging techniques.

A key goal of these traditional Cray supercomputer designs that use few (1-16) processors is to make each processor as fast as possible. But despite all the dramatic improvements in microchip and packaging technology, these kinds of supercomputer designs are reaching a physical limit—the speed of light. Computer signals travel through the computer's electrical circuitry at the speed of light. And no matter how much these computers with 1-16 processors speed up each processor, they would never attain some of the speeds that many compute-intensive jobs need [for example, supercomputers still cannot synthesize a protein from its gene] because of the physical limit imposed by the speed of light. This is where massively parallel computers (MPC) come in.

In massively parallel computers, hundreds or thousands of processors are put on one job, with each processor simultaneously tackling an assigned stage of the job to get the whole job done faster than one processor operating sequentially—the structure of the job permitting. So rather than trying to speed up one or a few processors to do the job, massively parallel computers (MPCs) put very many processors on the job to perform it in parallel. Now, the speed of light is no longer the physical limit, and execution of inherently parallel jobs can be speeded up considerably. Thinking Machines' CM5 uses

hundreds of 32-bit SPARC CMOS (Complementary Metal Oxide Semiconductor) microprocessors and runs at 128 GFLOPS peak. The physical limit to the speed of MPCs will eventually be the ability of the processors to communicate with each other.

MPCs use readily available CMOS (a proven technology) chips that consume less power than the ECL (Emitter Couple Logic) chips used in conventional (Cray-like) supercomputers, and these CMOS chips don't have to be as fast as ECL chips since it is not the speed of each one that matters (at this stage of the technology) in MPCs but their combination. And because they consume less power, they are air-cooled and don't need the elaborate liquid cooling systems of ECL-based systems.

MPCs can be divided into two groups: multiprocessors and multicomputers. Multiprocessor MPCs like Kendal Square Research's KSR 1 have numerous processors that share one memory bank. The KSR 1 has 1088 64-bit microprocessors that share the same memory bank. Multicomputer MPCs are interconnected microprocessors, each with its own memory, that communicate via message passing. Examples are Thinking Machines CM5, Intel's Paragon, and supercomputer MPCs from Ncube, Ametek, and Transputer.

Most of the manufacturers of traditional supercomputers (those with 16 or fewer very fast processors, and elaborate cooling systems) like Cray Research Inc., IBM, etc. have either already started MPC programs or announced that they will do so. But their quest to improve traditional supercomputers has not stopped. When, in 1989, Seymour Cray left Cray Research to start Cray Computer, his answer to getting a faster supercomputer was to use gallium arsenide chips which can be two and half times as fast as conventional silicon chips and also consume a lot less power. Gallium arsenide is a relatively new technology that is still in its infancy compared to the silicon semiconductor technology that now provides chips for computers. The introduction of the Cray-3 has been delayed primarily because of the difficulties in getting GaAs chips to work.

Supercomputer Systems Inc. (SSI), another supercomputer start-up is also having difficulties delivering its first supercomputer because it was banking on GaAs chips.

Another viable set of computers are the so-called minisupercomputers. They utilize the same vector processing of traditional supercomputers, but with some important differences: They are cheaper, provide 25 to 35 percent the performance of traditional supercomputers [Kelley 1988] , offer lower price for the performance provided and lend themselves to those low-end applications that don't need the higher performance of higher power supercomputers, let alone their prices. They use proven CMOS (Complementary Metal Oxide Semiconductor) chips that are less expensive and consume less power than the power-demanding but faster ECL chips used in traditional designs. This results in cheaper systems that are air-cooled.

2.2 The Hypercube Model and Supercomputers

In this section we use the hypercube of innovation model to examine the supercomputer industry that we have just described. However, this is not a comprehensive treatment of innovations in supercomputers.

Table 12 lists key supercomputer innovations, while their impact on the capabilities and assets of their innovators, suppliers, customers, and complementary innovators is shown in Table 13 and figure 9.

Using the Henderson and Clark categorization criteria, vector processing was an architectural innovation to CDC (Star-100) and Cray Research (Cray-1) when they designed these systems. The main components of the supercomputer—memory, CPU, I/O—and core design concepts had not changed radically; the key change was the provision of vector processing. But the linkages between these components and core concepts were being altered. To many customers and suppliers of applications software, however, this was a radical innovation because they had to learn how to program with vector processors. Luckily, for the Cray, most users of supercomputers then were scientists

who wrote their own software and were more interested in a number-crunching engine than a complete data processing solution. This a relatively small segment of the market that would remain small for this very reason. More importantly, a complementary innovation, the vectorizer was developed that could convert some of the old software written for scalar machines to forms in which vector machines could crunch. So the impact of the radicalness of the innovation on customers was not that important. Cray Research's multiprocessor, Cray X-MP, can also be considered an architectural innovation for reasons similar to those just listed above. For customers, its impact was more incremental than radical for several reasons since it still used the same Cray Operating System (COS).

The Cray-3 and SSI's machine are examples of machines that are radical to suppliers and facing problems because of it. Both machines are multiprocessor but with no more than 16 processors and not radically different from previous designs. They are, however, depending on GaAs chips to make major contributions to the planned speed improvements. But GaAs technology is still in its infancy compared to the proven silicon technology that other computers use and is thus a radical innovation to any computer. Cray Computer's solution to reducing this uncertainty was to acquire Gigabit Logic, a GaAs chip manufacturer. That has still not worked. While the problems with the Cray-3 and SSI's machine may not be entirely due to GaAs chips, it is true that GaAs, a radical innovation to most suppliers of chips, has contributed to the problems of the two machines.

Massively parallel computers (MPCs) are a radical innovation for all members of the innovation value-added chain except suppliers. Their design is conceptually very different from that of traditional supercomputer designs. Writing software for them is even trickier. Users of the installed base of traditional Cray-type designs would prefer machines that allow them to keep some of the skills and knowledge acquired with the Cray-like machines, and especially any applications programs that they may have written. Their operating systems are also different. Applications, as well as systems programmers

for the new machines are also not easy to find. Hardcore supercomputer users (scientists and academics) can write their own software. But for MPCs to diffuse into general purpose applications that will greatly increase their success, they need lots of software. In particular, MPCs need to be programmable in existing programming languages like FORTRAN, C and C++. This would mean that the end-user only sees the change in speed.

Section IV: Summary and Implications of the Hypercube.

Using several examples, we have shown that an innovating entity that only looks at the impact of its innovation on its competence and existing products, and does not critically examine the impact of that innovation on the competence and capabilities of its suppliers, customers, competitors and complementary innovators, may be making a mistake. Dvorak's DSK keyboard failed to diffuse because it was an architectural innovation to Dvorak but a radical innovation to its customers. OS-2 was a radical innovation to IBM but an incremental innovation to DOS users versus Microsoft Windows which is an incremental innovation to Microsoft and an incremental innovation to users of DOS. This allowed Microsoft to enter the market early and so far, Microsoft Windows is winning. Similarly, Lotus, that didn't pay attention to the momentum of Windows software, lost some ground in its spreadsheet market share to Microsoft's Excel. The case of the electric car—which is a radical innovation to the innovating firms, suppliers of components and complementary products, but an incremental innovation to users—was also discussed.

The model forces innovation managers to look at their innovations not only in terms of what the impact of the innovation will be on the innovating entity's capabilities and assets, but also those of suppliers, customers, and complementary innovators. We suggest that innovators should think twice about innovations that destroy skills, knowledge and positive network externalities at any of the stages of the value-added chain, especially

at the customer level. They should avoid the Red Zone (of the mapping of innovations along the innovation value-added chain) and go with innovations that reinforce key concept and linkages all along the value-added chain (innovations that fall in the green or yellow zone).

We also note some criteria for innovating in the Red Zone. Specifically, we suggest that the Red Zone should be avoided unless a subset of the following is true: 1) The price/performance ratio of the innovation, as viewed by all the levels of the value-added chain especially the customer, outweighs any losses incurred as a result of competence or positive network externality destruction. This happens for example when the physical limit of an older technological trajectory has been reached and the only way to overcome this physical limitation is to move to a new technological trajectory—a move that often means destruction of competence acquired during the evolution along the older trajectory but great improvement in some key parameter. 2) New markets where customers have not yet had time to build any innovation-specific skills and knowledge, and competence destruction is not an issue. 3) Complementary innovations, that allow customers (or other members of the innovation value added chain) to keep their competence and positive network externalities, exist. 4) When institutional requirements mandate the innovation.

We analyzed RISC and CISC chips, and supercomputers using the model. In particular, we analyzed the impact of key innovations in CISC (Complex Instruction Set Computers) chips, RISC (Reduced Instruction Set Computers) chips, and supercomputers on the capabilities of suppliers, customers, and complementary innovators. In CISC, we suggested that Intel's foresight in designing the 8088 microprocessor in response to the inertia of complementary 8-bit chips, may have contributed to its being chosen by IBM over competitors to provide the microprocessor architecture for the now very popular IBM PC and PC compatibles. This may have been Intel's most important decision ever.

We also found that although RISC is an architectural innovation as far as chipmakers like Motorola and Intel are concerned, it is a radical innovation to OEM customers who have been using CISC chips to design personal computers and sell to end users. This is because with RISC, these OEMs have to learn new assembly languages, establish new development systems, retrain their engineers on how design systems with the RISC chips. To personal computer end-users who have learned to use DOS, acquired or written their own applications programs, and established positive network externalities on CISC-based machines, RISC is a radical innovation since in its present form, it destroys the competence, capabilities and positive network externalities of these customers. The promise of speed alone is not enough to dislodge CISC in this particular market. For computer systems under $25,000, annual sales of Intel's CISC chips alone exceed 20 million units while all RISC chips combined had sales of only 339,000 units (Business Week, March, 30, 1992)[2]. We also suggest that it may be the realization of the inertia of CISC vis-a-vis RISC that made firms like Compaq pull out of ACE consortium. All that could change if a complementary innovation (e.g. software) could be developed that allows all DOS users to preserve their skills and old applications software when they use RISC machines. Microsoft NT is intended to be this innovation.

In newer markets like Workstations where the capabilities, competencies, and positive network externalities have not been well-established yet, RISC is doing very well. In the embedded control market where speed is critical and the end-user is not locked into CISC as in the PC market, RISC is also doing well.

In the Minicomputer and Mainframe markets the price/performance advantages of RISC have been sufficiently compelling that manufacturers and customers of these classes of computers have adopted RISC/UNIX technology instead of the mostly CISC/proprietary operating system solutions of the past. There are still, however, many manufacturers of proprietary systems.

---

[2] Our thanks to an anonymous referee who suggested this example.

In supercomputers, Cray Computer Corp. and SSI are having difficulties introducing their new gallium arsenide (GaAs) chip-based supercomputers partly because GaAs is a radical innovation to chip suppliers relative to the mature silicon technology.

Earlier versions of supercomputer innovations that were radical innovations at customers didn't have the disastrous consequences predicted by the hypercube model because many of those early users were scientists and academics who wrote their own programs, and could trade the program writing for a more powerful computing engine.

Massively parallel computers, despite being faster than the traditional Cray-like supercomputers may not be diffusing as fast as one would expect because they are a radical innovation not only to the innovating entities but also to customers and suppliers of complementary innovations like software. It is, however, an incremental innovation for suppliers of hardware components like microchips and disk drives.

The real breakthrough in supercomputer diffusion will come when the parallel machines penetrate the general purpose business applications that could use their compute power. This will come only if the software is there, which in turn, can only be developed if the current parallel machines can be programmed with existing languages such as FORTRAN, C, C++, etc.

Conclusion

The common practice of classifying innovations only according to the impact of the innovation on the innovating entity's capabilities vis-a-vis its existing technology and markets is not adequate for high technology products that require critical input components and equipment from suppliers, depend on complementary innovations for success, require high levels of learning by customers before use, and that lend themselves to positive network externalities. For such products, the impact of the innovation on the capabilities and assets of suppliers, customers and complementary innovators may be just as critical as that on the innovating entity's competence and assets.

The hypercube model forces managers at the innovating entity to evaluate their innovations in terms of the impact of those innovations on the competence and assets of all the members of the innovation value-added chain. Our examination of the CISC, RISC and supercomputer industries suggests that the innovator should pursue innovations that reinforce core concepts and competence along the innovation value-added chain, while being more cautious with those that don't. The innovator should watch out for the inertia of older complementary innovations and the momentum of newer ones, and take advantage of them.

Areas of future research:

We have developed the hypercube model using qualitative data. An area of further research would be the collection of quantitative data to further gain insights to this concept.

The second and equally important area of research would be the extension of the hypercube to include multi-innovator/competitor scenarios. Such an expanded model could then be applied to innovation-based international competition.

References:

Abernathy, W. J. and Clark, K., Mapping the Winds of change, *Research Policy* (1985) 14 3-22.

Abernathy, W. J. and J. M Utterback, Patterns of innovation in technology, *Technology Review* (1978) 80:7 (June-July) : 40-47

Afuah, A. N. and J. M Utterback, The Emergence c' a New Supercomputer Architecture, *Technology Forecasting and Social Chang*: 1991) 40, 315-328

David, P. A., A Contribution to the Theory of Innovation. Memorandum No. 71, Stanford Center for Research in Economic Growth. Stanford University 1969

David, P. A., Clio and the Economics of QWERTY, *American Economic Review: Papers and proceedings* (1985) 75: 2, 332-6

Dobbs, C., Reed, P. and Ng, T., Supercomputing on a Chip - Genesis of the 88000, *VLSI System Design*, 1(988) May, pp. 24-33.

Els, Sharon A., Configuration and Competitiveness in the Supercomputer Industry." Master's Thesis at MIT. 1989

Farrell, J., and Saloner, G., Standardization, Compartibility and Innovation. *Rand Journal of Economics* (1985) 16: 1, 70-83.

Fernbach, S., Supercomputers--Past, Present, Prospects, *Future Generation Computer Systems*, July, 1984.

Foster, R.A.,. "*Innovation : The attacker's Advantage.*" (Summit Books, N.Y., 1986)

Freeman, C., *The Economics of Industrial Innovation* (MIT Press. Cambridge 1986. Second Edition)

Gill, J., Gross, T., Hennessy, J., Jouppi, N., Przybylski, S. and Rowen, C., *Summary of MIPS Instructions*, (1983) Stanford University Technical Note no. CSL-83-237.

Henderson, R. and CLark, K., Architectural Innovation:The reconfiguration of existing product technologies and the failure of established firms, *Administrative Sciences Quarterly* (1990) 35, 9-30

Henderson, R., Underinvestment and incompetence as responses to radical innovation: Evidence from the photolithographic alignment equipment industry, *RAND Journal of Economics* (1993) Vol. 24, No. 2, Summer 1993

Hewlett-Packard, Precision Architecture and Instruction Reference Manual (Hewlettt-Packard Company, 1986).

Kane, G., *MIPS R2000 RISC Architecture* (Prentice Hall, Englewood Cliffs, N.J. 07632, 1987)

Kartz, Michael L., and Shapiro, C. (1985): Network Externalities, Competition and Compartibility. *American Economic Review* 75:3, 424-40.

Kartz, Michael L., and Shapiro, C. (1986): Technology Adoption in the Presence of Network Externalities. *Journal of Political Economy* 94: 4, 822-41

Katevenis, M.G.H., *Reduced Instruction Set Computer Architectures for VLSI* (MIT Press, Cambridge, MA 1985)

Lunde, A., Empirical Evaluation of Some Features of Instruction Set Processor Architecture, *Communications of the ACM* (1977) 20, no. 3, pp. 143-153

Patterson, D. and Sequin, C., RISC I: A Reduced Instruction Set VLSI Computer, *Prooceedings of the 8th Annual Synposium on Computer Architecture*, (1981) ACM SIGARCH CAN 9.3, pp. 443-457.

Patterson, D.A. and Ditzel, D.R., The Case for the Reduced Instruction Set Computer, *Computer Architecture News* (1980) 8 no. 6, pp. 25-33.

Pratt, G. A., Electric Cars: On the Road again, *Technology Review* August/September (1992).

Radin, G., The 801 Minicomputer, *IBM Journal of Research and Development* (1983) 27, no. 3, pp. 237-246.

Reinganum, J., Uncertain Innovation and the Persistence of Monopoly, *American Economic Review*, Vol. 73 (1983) 741-748

Reinganum, J., Uncertain Innovation and the Persistence of Monopoly: Reply, *American Economic Review*, Vol. 74 (1984) 243-245

Roberts, E. and Berry, C.A, Entering New Businesses. *Sloan Management Review* (Spring, 1985) 3-17

Schumpeter, J. A, The Creative Response in Economic History, *Journal of Economic History* 7 (1947), 149-159. Reprinted in *Essays of J. A. Schumpeter*, ed. by R. V. Clemence, Cambridge, MA.: Addison-Wesley, 1951.

Sun Microsystems, *The SPARC Architecture Manual* Sun Microsystems Inc., (1987) Mountain View, California.

Teece, David J., Profiting from Technological Innovation: Implications for Integration, Collaboration, Licensing and Public Policy, *Research Policy* (1986) 285-305

Tushman, Michael L., and Anderson, P., Technological Discontinuities and Organizational Environments." *Administrative Science quarterly*, 31 (1986): 439-465

Utterback, J. M., and Kim, L., Invasion of a Stable Business by Radical innovation, *The Management of Productivity and Technology in Manufacturing*, by Kleindorfer, P R. (Plenum Press 1986)

Utterback, James M., Innovation and Industrial Evolution in Manufacturing Industries." in *Technology and Global Industry: Companies and Nations in World Economy* edited by Bruce R. Guile and Harvey Brooks. National Academy Press. Washington D. C. 1987. To be elaborated on in *The Dynamics of Innovation in Industry.* Chapter 5. Harvard Business School Press, 1994

von Hippel, E., *The Sources of Innovation* (New York, Oxford University Press, 1988)

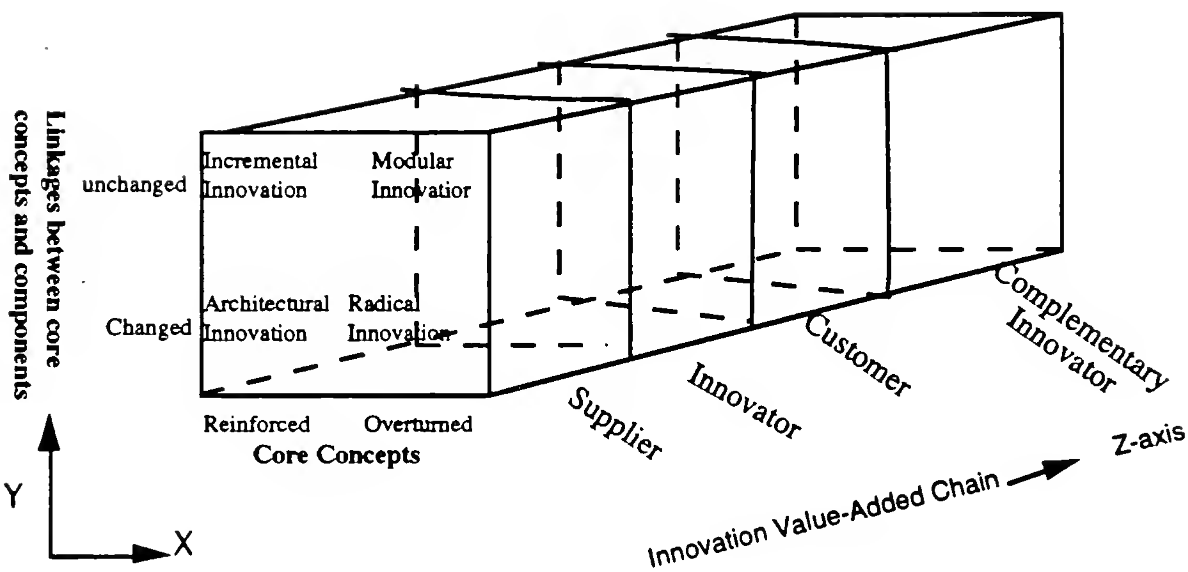


Figure 1: The Hypercube of Innovation. The X and Y axes are the innovation-classifying factors. The Z-axis is the innovation value-adding chain of supplier of key components, innovator, Customer and supplier of complementary innovators

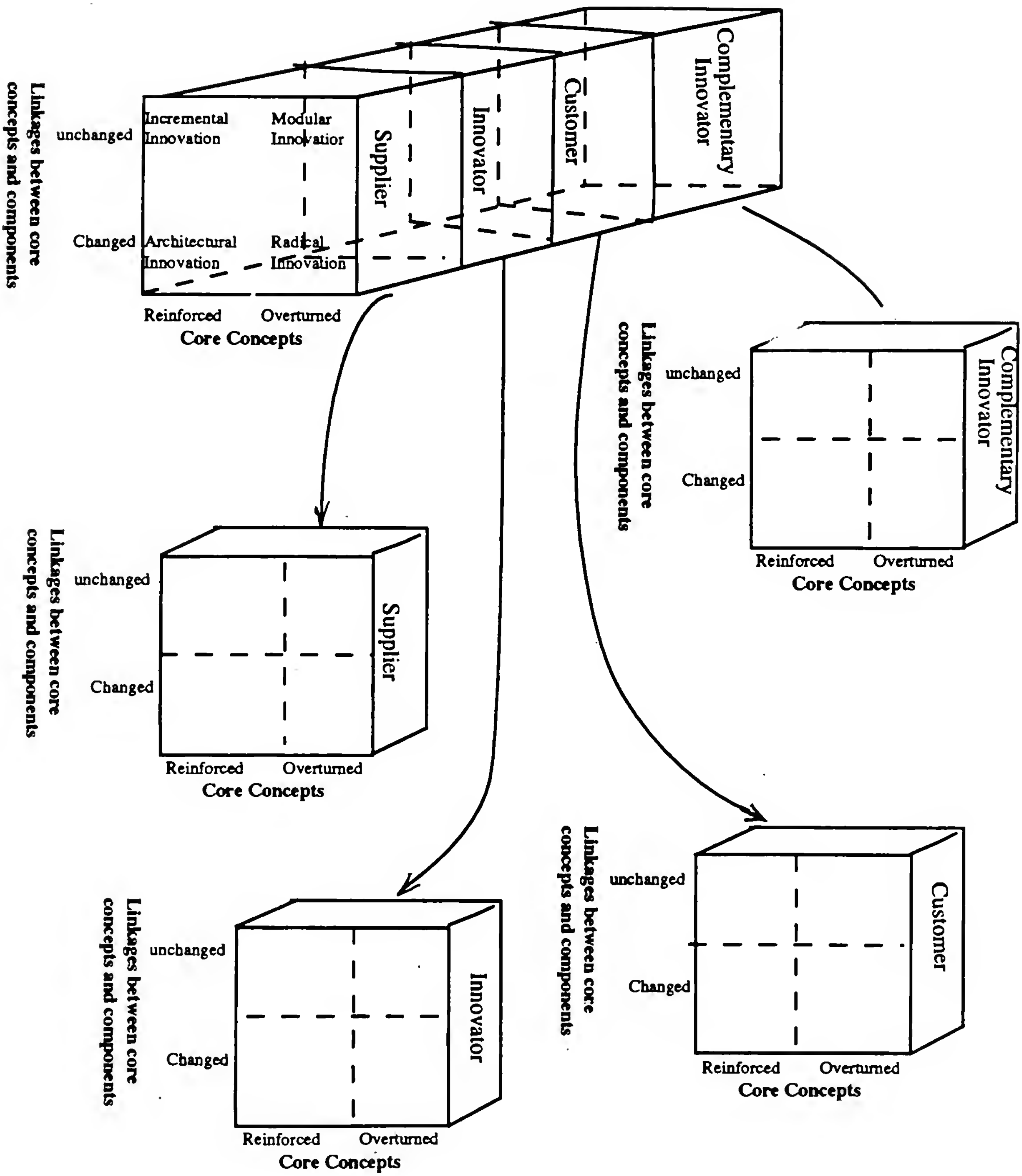


Figure 2: The hypercube of Innovation exploded to show the various faces that an innovation can assume along the innovation value-added chain.

FIGURE 3): The GREEN-RED zone map

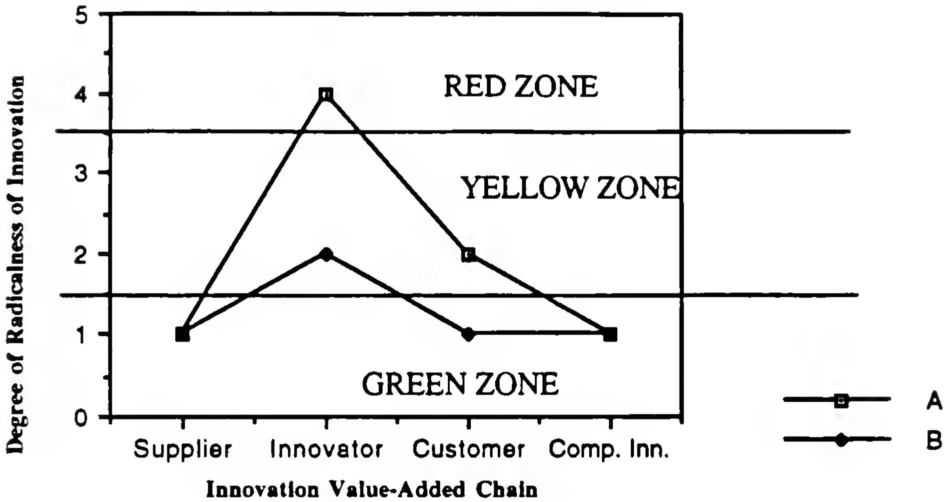


FIGURE 4a): The GREEN-RED zone map for the DSK keyboard.

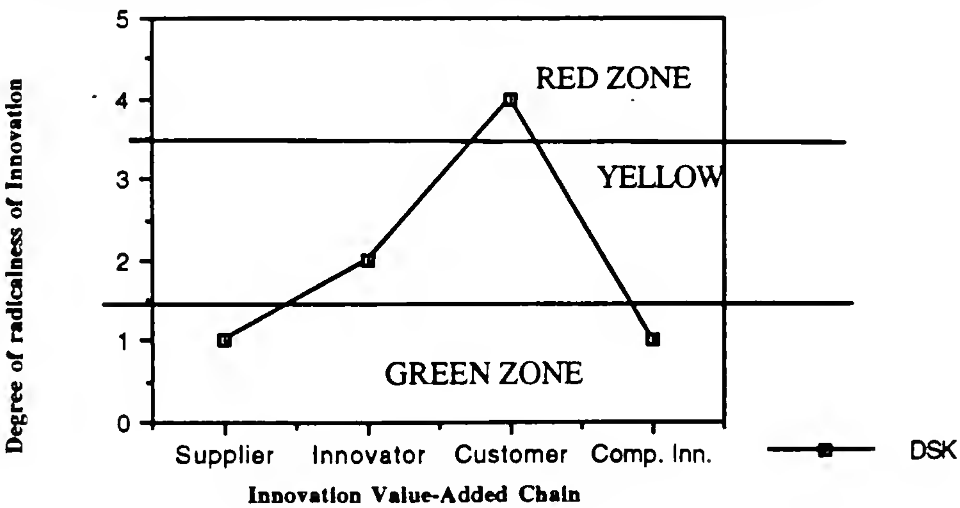


FIGURE 4b): The GREEN-RED zone map for the Electric Car.

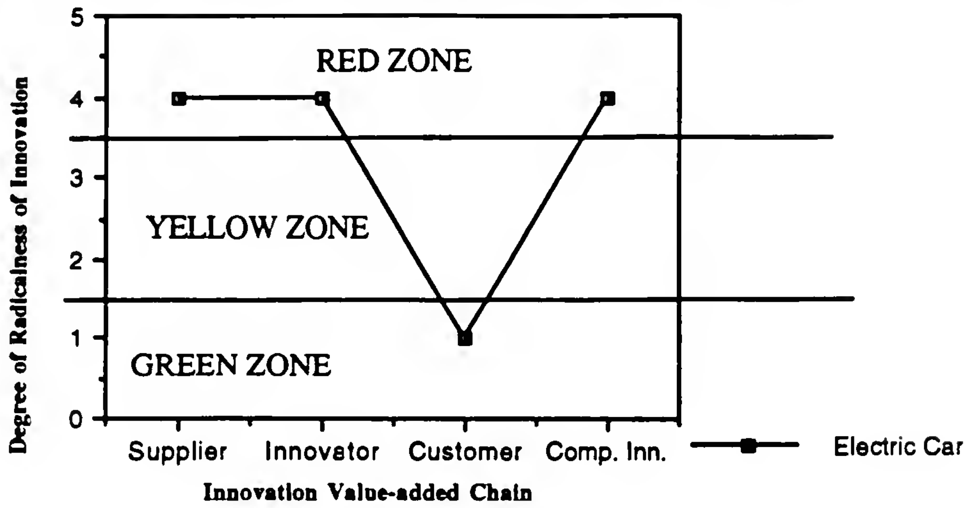
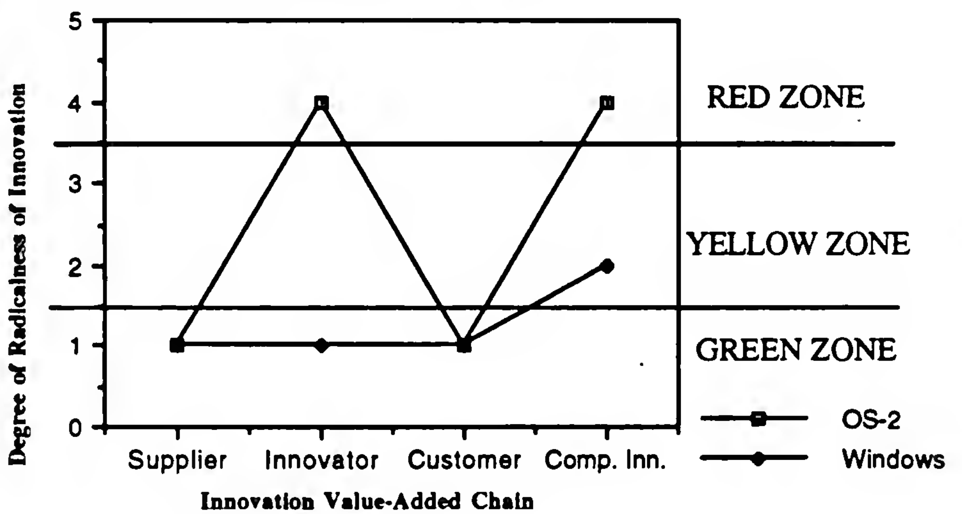


FIGURE 5): The GREEN-RED zone map for IBM's OS-2 and Microsoft's Windows



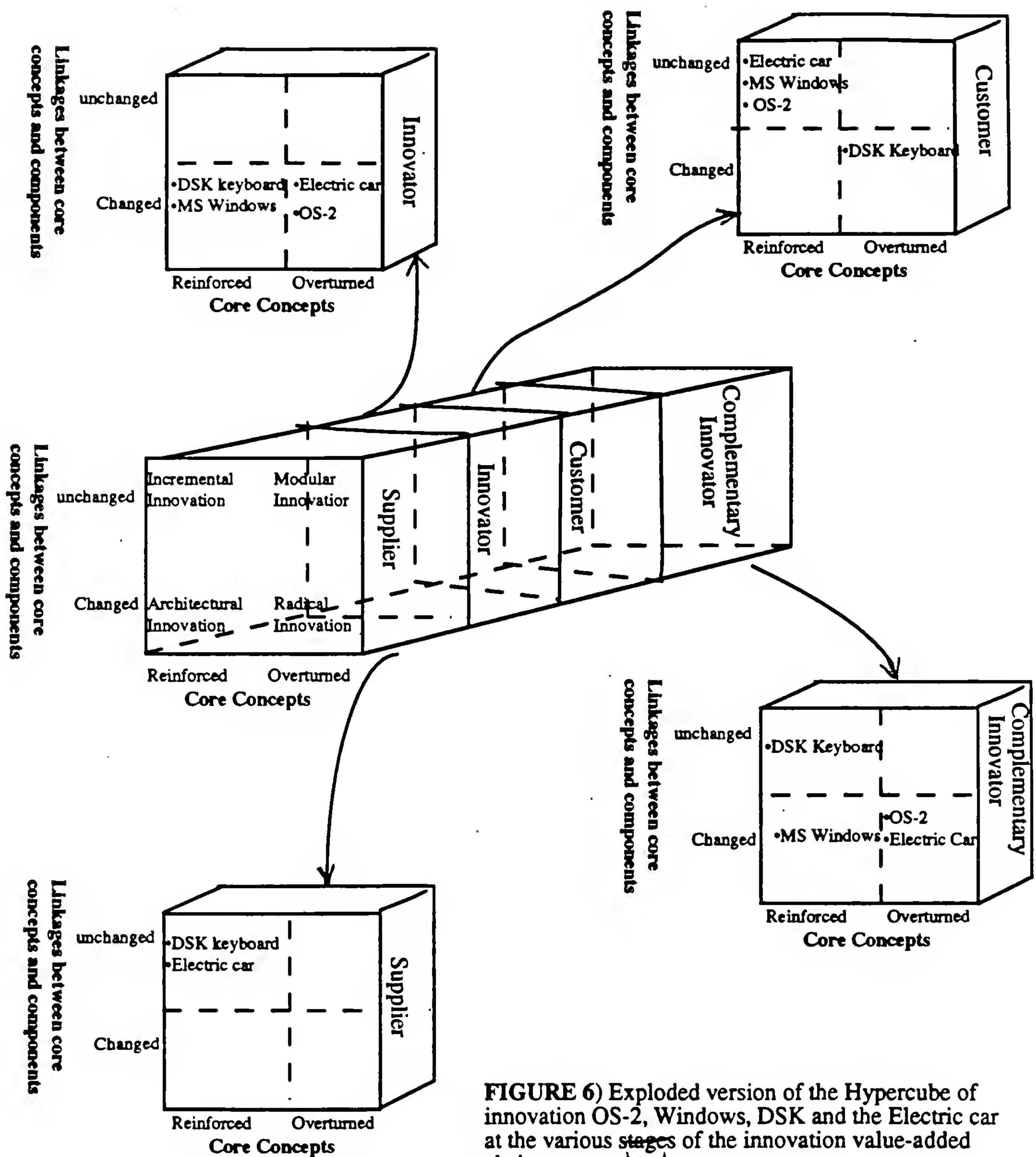


FIGURE 6) Exploded version of the Hypercube of innovation OS-2, Windows, DSK and the Electric car at the various ~~stages~~ levels of the innovation value-added chain.

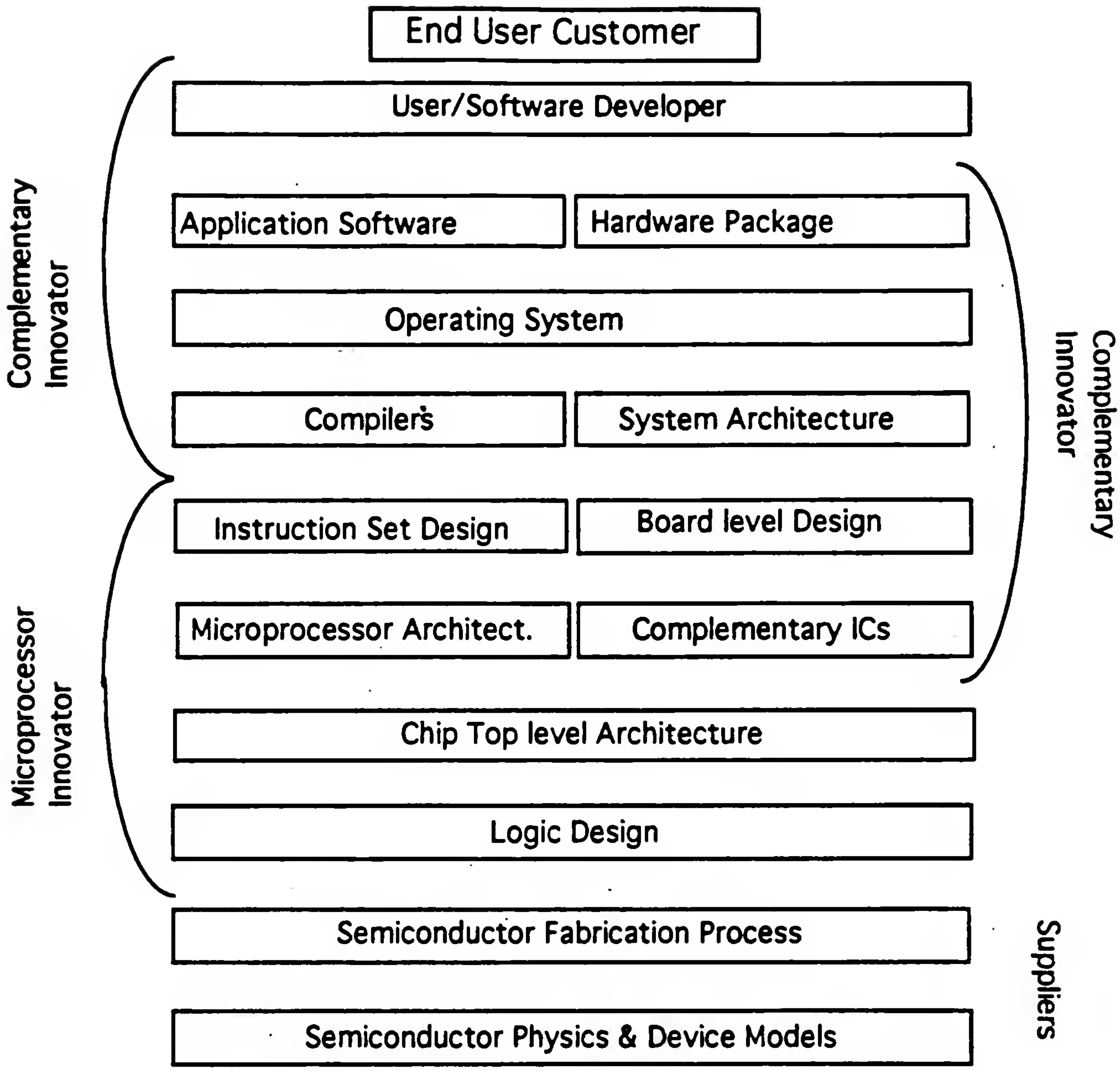


Figure 7. Computer System knowledge Areas and their providers .

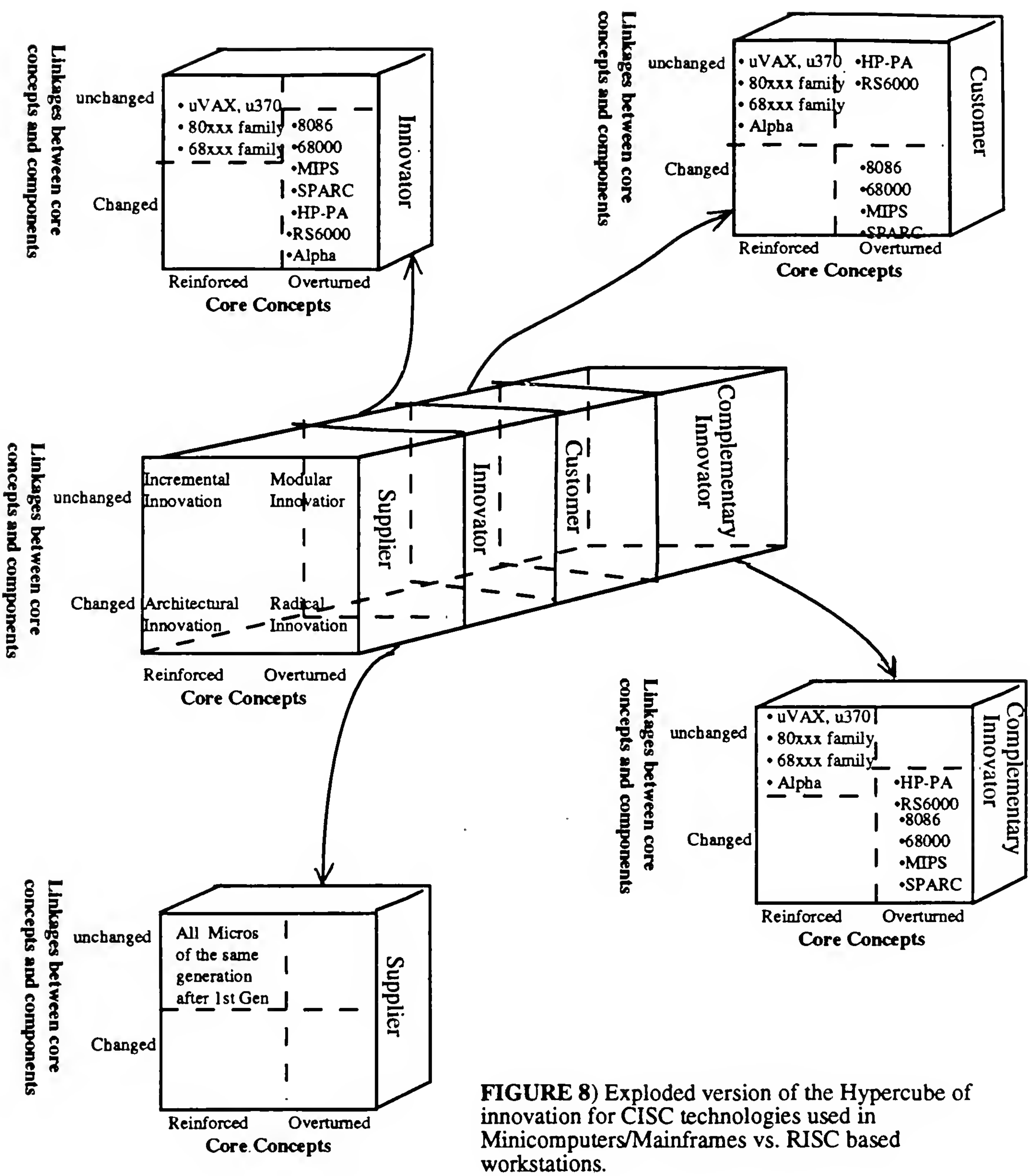


FIGURE 8) Exploded version of the Hypercube of innovation for CISC technologies used in Minicomputers/Mainframes vs. RISC based workstations.

Table 2: The Hypercube in tabular form—CISC & RISC innovations

INNOVATOR	CUSTOMER
Incremental • uVAX, u370 (CISC) • 80xxx family(CISC) • 68xxx family(CISC) ***Radical*** • 8086(CISC) • 68000(CISC) • MIPS R2000(RISC) • SPARC1(RISC) • HP-PA (RISC) • RS6000(RISC) • ALPHA(RISC)	***Incremental*** • uVAX,u370 (CISC) • 80xxx family (CISC) • 68xxx family (CISC) • Alpha ***Radical*** • 8086(CISC) • 68000(CISC) •MIPSR2000(RISC • SPARC1(RISC) • HP-PA (RISC) • RS6000(RISC)
SUPPLIER	**COMP. INNOVATOR**
Incremental All micros of the same generation ***Radical***	***incremental*** • uVAX,u370 (CISC) • 80xxx family (CISC) • 68xxx family (CISC) • Alpha ***Radical*** • 8086(CISC) • 68000(CISC) •MIPSR2000(RISC • SPARC1(RISC) • HP-PA (RISC) • RS6000(RISC)

Table 10: Some supercomputer product innovations.

Machine	Year	Mfr	Bits	CPU	Technology	O.S
IBM700	1954	IBM	36	Sequential	Vacuum T	
IBM7000	1959	IBM	36	Sequential	Transistor	
CDC6600	1965	CDC	60	Sequential		
CDC7600	1969	CDC	60	Scalar	Transistor	
Star-100	1972	CDC	64	Vector Proce	ICs	
ILLIAC IV	1972	Burroughs	64	64 Process	ICs	
Cray-1	1976	CDC	64	Vector/Scalar	LSI	
CDC 205	1976	CDC	64	Vector	LSI	
Univac LARC	1960	Univac				
IBM7030	1960	IBM				
IBM360/195	1971	IBM				
TIASC	1974	T I	32	Vector	LSI	
Denelcor HEP-1	1977					
Cyber205	1981	CDC	64	Vector	LSI	
Hitachi S810 20	1983			Vector/Scalar	LSI	
Fujitsu VP				Vector		
ETA-30						
Cray X-MP	1982	Cray	64	Vector.		1st Multicprocesso
Denelcor HEP-1		Denelcor		Multiprocessor		
ETA-10		ETA Systems		Vector. 8 processors		
Cray-2	1985	Cray	64	4 CPUs		
Cray Y-MP	1988	Cray		8-16 CPUs Vector		UNICOS, COS, CTSS
Cray C-90	1991	Cray		Vector. 16 processors	ECL	UNICOS, COS, CTSS
Cray-3					GaAs	UNICOS, COS,
IBM3090/600S VF	1988	IBM		Vector. 1-6		MVS, AIX, VM/CMS
Fujitsu VP-2600/20	1991	Fujitsu		Vector		Propr. OS. UTS/M
Fujitsu FACon VP-200	1984	Fujitsu			ECL	
NEC SX-2	1985	NEC			ECL	
Hitachi S820/80	1988	Hitachi		Vector	ECL	Proprietary OS, HIUX
NEC SX-3	1992			Multiprocessor . Vector		UNIX

Table 12a: Key innovations in supercomputers

Innovation	Machine	Year	Firm
Vector Processing	Star-100	1973[3]	CDC
Vector processing	Cray-1	1976	Cray Research
Multiprocessing (traditional)	Cray X-MP	1982	Cray Research
MPC SIMD	CM-2	1986	Thinking Machines
MPC multiprocessor	KR 1	1992	Kendal Square Research
MPC multicomputer	Intel Paragon	198x	Intel Corp
Minisupercomputers	Convex-2	198x	Convex Computers
	TIASC	1974	TI
	Denelcor HEP-1	1977	Denelcor
	Cyber205	1981	CDC
	Hitachi S810 20	1983	Hitachi
	Fujitsu VP		Fujitsu
	Cray-2	1985	Cray
	Cray Y-MP	1988	Cray
	Cray C-90	1991	Cray
	Cray-3		
	IBM3090/600S VF	1988	IBM
	Fujitsu VP-2600/20	1991	Fujitsu
	Fujitsu FACon VP-200	1984	Fujitsu
	NEC SX-2	1985	NEC
	Hitachi S820/80	1988	Hitachi
	NEC SX-3	1992	NEC

Table 12b: Classifications of some of Cray Research's Innovations:

Year	Product	Innovation	Operating System	Microchip Technology	Cray Research
1976	Cray-1	Vector Processing	COS (Cray Op System)	ECL	radical Innovation
1979	Cray-1/S		COS		
1982	Cray-1/M		COS	MOS memory	
1982	Cray X-MP	Multiprocessi ng	COS		
1985	Cray-2	4-CPUs	UNICOS (instruction set different)		
1988	Cray Y-MP	8-CPUs			
1990	Cray Y-MP 2E	air/water-cooled			
1991	Cray Y-MP 8E				
1991	Cray Y-MP		UNICOS		
199x	Cray-3				

[3] designed in 1969 but became operational in 1973. Never shipped. Features maintained in the Cyber 205. (1982)

Table 13: The Hypercube in tabular form—Supercomputer Innovations

INNOVATOR	CUSTOMER
Incremental • Cray Y-MP 2E • Cray Y-MP 8E • Cray-1 • Star-100 • Cray X-MP • Cray-2 • Convex-2 • Cray Y-MP C90 • Cray-3 • SSI *Radical* • Illiac IV • CM-2 • Paragon • KR1 • CM-5	*Incremental* • Cray X-MP • Cray-2 • Convex-2 • Cray Y-MP C90 *Radical* • Illiac IV • Cray-1 • Star-100 • CM-2 • CM-5 • Paragon • KR 1
SUPPLIER	**COMP. INNOVATOR**
Incremental • Illiac IV • Cray-1 • Star-100 • Cray X-MP • Cray-2 • Convex-2 • Cray Y-MP C90 • CM-2 • CM-5 • Paragon • KR 1 *Radical* • Cray-1 • SSI	*Incremental* • Cray X-MP • Cray-2 • Convex-2 • Cray Y-MP C90 *Radical* • Illiac IV • Cray-1 • Star-100 • CM-2 • CM-5 • Paragon • KR 1

Table 14: The Hypercube in tabular form—Cray Research Inc.

INNOVATOR	CUSTOMER
Incremental • Cray Y-MP 2E • Cray Y-MP 8E • Cray-1 • Cray X-MP • Cray-2 • Cray Y-MP C90 • Cray XMS ***Radical***	***Incremental*** • Cray X-MP • Cray-2 • Cray Y-MP C90 • Cray XMS ***Architectural*** ***Radical*** • Cray-1
SUPPLIER	**COMP. INNOVATOR**
Incremental • Cray-1 • Cray X-MP • Cray-2 • Cray Y-MP C90 • Cray XMS ***Radical*** • Cray-1	***Incremental*** • Cray X-MP • Cray-2 • Cray Y-MP C90 • Cray XMS ***Radical*** • Cray-1

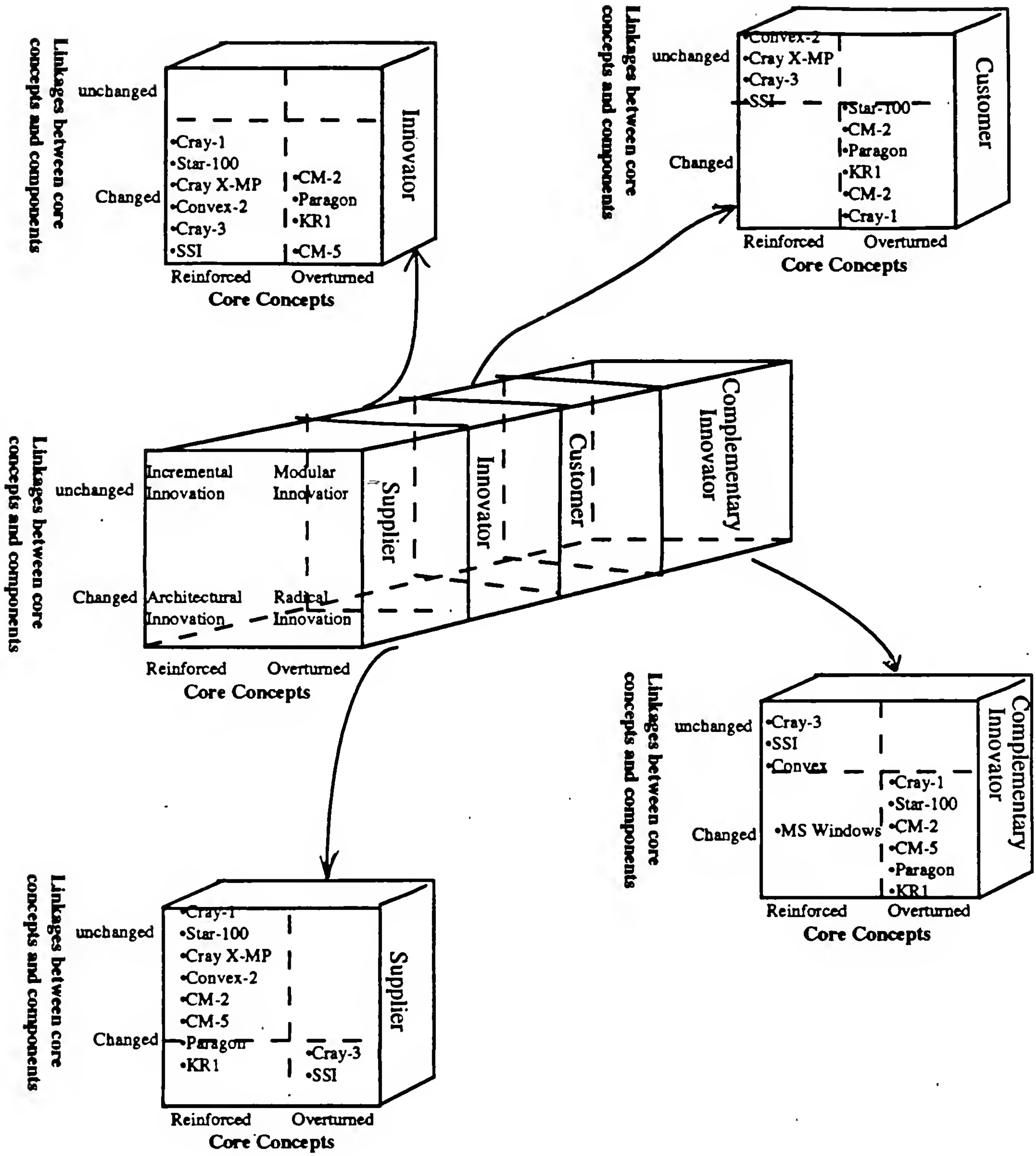


Figure 9 :Supercomputers and the Hypercube

2935 45

MIT LIBRARIES
3 9080 00856464 0

Date Due

Lib-26-67